SHARDUS_SOFTWARE_LICENSE_2023-02-02
Replaces version 2022-07-27 0x50809cac4f8ffdc62a6a016a79656fedbaeb1e70b6fbbb6b7acd9b8556a0572c

Copyright © 2017 - Present   Shardus Association

1. The latest version of this license file can be obtained from https://shardus.com/license.
2. The hash (unique signature) of this document will be notarized on the blockchain and a link to the transaction will be posted on the license web page mentioned above.
3. The original and most current version of the Shardus software is available from https://GitLab.com/shardus.
4. Older versions and modified versions of the Shardus software are also referred to in this document as Shardus software.
5. If you did not download the Shardus software from the original source then you may have received it as part of another software package which includes the Shardus software.
6. This license file must be included without changes along with any distribution of the Shardus software.
7. THE SHARDUS SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SHARDUS SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SHARDUS SOFTWARE.
8. The parent license of any software that includes the Shardus software does not apply to the Shardus software. Only this license file defines the terms for the Shardus software.
9. Although the Shardus software may be open source the use of the Shardus software is subject to the following terms.

10. For Non-Commercial Use:
    a. The Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License is used. https://creativecommons.org/licenses/by-nc-sa/4.0/legalcode
        i. You are free to:
            1. Use — run this software for testing, personal, research or other non-commercial use
            2. Share — copy and redistribute the software in any medium or format
            3. Adapt — remix, transform, and build upon the software
    b. The licensor cannot revoke these freedoms as long as you follow the license terms:
        i. Attribution — You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
        ii. NonCommercial — You may not use the software for commercial purposes.
        iii. ShareAlike — If you remix, transform, or build upon the software, you must distribute your contributions under the terms of this license and include a copy of this license file in your distribution.
        iv. No additional restrictions — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.
    c. Limitations:
        i. This license is non-exclusive meaning the author may grant this license to others.
        ii. This license may not be sublicensed or relicensed meaning that you may not apply any other license to this software.
        iii. This license grants privileges only for the software and no other intellectual property such as patents, trademarks or copyrights.
    d. Notices:
        i. You do not have to comply with the license for elements of the software in the public domain or where your use is permitted by an applicable exception or limitation.
        ii. No warranties are given. The license may not give you all of the permissions necessary for your intended use. For example, other rights such as publicity, privacy, or moral rights may limit how you use the software.

11. For Commercial Use:
    a. When the Shardus software is used in development or testing this is considered non-commercial use and the non-commercial license still applies.
    b. When the Shardus software is used in production the Licensee must take further action to initiate the process of obtaining a commercial license within 30 days of beginning production use.
    c. The Licensee must use the most current version of the license file to determine the requirements for obtaining a commercial license. These requirements are different based on a public or private network.
    d. For Public Networks
        i. With a coin, token or other asset
            1. The launch of a public mainnet marks the beginning of production use.
            2. The network must be launched with rules to distribute at least 1% of the maximum or inflating supply to Shardus token (ULT) holders through a claim process. The details of the process is described below in the Claim Process section.
        ii. With no coin, token or other asset
            1. Same requirements as a private network.
    e. For Private Networks
        i. The launch of an internal or federated product of service based on a private network using the Shardus software marks the beginning of production use. Or for public networks with no coin, token or other asset the launch of a public mainnet marks the beginning of production use.
        ii. The Licensee must acquire Shardus tokens and convert them to a license token on the Ethereum network. The Shardus smart contract on the Ethereum network has a "purchaseLicense" function which burns the Shardus tokens and issues a license token to the sender. This process is described below in the Purchase License section.
    f. The commercial license applies only to one network. Additional production networks will need to obtain their own separate commercial license even for the same entity.
    g. The conditions of the Creative Commons license do not apply once a commercial license is obtained. Instead:
        i. You are free to:
            1. Use — run this software for a commercial purpose
            2. Share — copy and redistribute the software in any medium or format
            3. Adapt — remix, transform, and build upon the Shardus software
        ii. You are not required to give Attribution.
        iii. You are not required to Share-Alike and may use your own parent license that applies to your software which includes the Shardus software. This license file must be included with the Shardus software and only this license file defines the terms for the Shardus software.
        iv. Limitations:
            1. This license is non-exclusive meaning the author may grant this license to others.
            2. This license may not be sublicensed or relicensed meaning that you may not apply any other license to this software.
            3. This license grants privileges only for the software and no other intellectual property such as patents, trademarks or copyrights.
    h. Once a Licensee has fulfilled the requirements to obtain a commercial license, the license is irrevocable.
    i. Although this license file may be updated in the future, previous versions of the license file will be honored for any Licensee that has already obtained a license based on a previous version of this file.
    j. The following users are not considered Licensees and do not need to obtain a commercial license if they were not involved with the initial creation or launch of the network:
        i. Any developer of decentralized apps on a smart contract platform based on the Shardus software.
        ii. Any operator of a node in a network based on the Shardus software.
        iii. Any developer contributing to the application software which includes the Shardus software. However, if these developers are involved with the creation or launch of the network then they are responsible to ensure that the rules to allow a distribution to the Shardus token holders as defined by this license are followed.

iv. Any user, developer or entity building on or using a network that has already obtained a commercial license for the Shardus software.

12. Claim Process
    a. This only applies to public networks with an asset.
    b. The actual implementation of the claim process may vary. The following provides guidelines on how it could be done. However, items which are noted as "must" are required.
    c. There must be a record of the claim process on a public blockchain. This must be done through a smart contract that allows claim transactions as defined later.
    d. A snapshot must be taken of all addresses holding Shardus token (ULT) as defined later.
    e. The Licensee must notarize the unique signature of the current license file on the Ethereum or Polygon blockchain using a key belonging to the Licensee. Instructions on how to do this are provided on the Shardus license page https://shardus.com/license/. This will ensure that even if the license file is changed the Licensee only needs to fulfill the requirements of the license file version that was notarized.
    f. Licensee must create an announcement page on the Licensee website:
        i. The announcement page must be created at least 30 days in advance of the first snapshot date or no later than 30 days after the public mainnet launch whichever is earlier.
        ii. The page must include the link to the block explorer transaction showing the hash of the license file the Licensee has notarized using a public key belonging to the Licensee.
        iii. The page must link to the smart contract used to process claims. The activity on this smart contract provides proof on the blockchain of fulfilling the requirements to obtain a license.
        iv. The page must state what percentage of the maximum or inflating supply of Licensee network coins, tokens or other assets are allocated to be claimed by Shardus token (ULT) holders. A minimum of 1% is required.
        v. The page must state the exact date and time of the snapshot.
        vi. The first snapshot must be within 100 calendar days of the mainnet launch.
        vii. For assets with a fixed or deflationary supply the Licensee may choose to distribute the allocated percentage of assets through multiple claim events. For example if 5% of the max supply of coins on the Licensee network will be given to ULT holders then 5 claim events can distribute 1% each.
            1. For each claim event the following must be specified when the announcement page is created:
                a. The snapshot date and time.
                b. The claim start date which must be within 7 days of the snapshot date.
                c. The claim end date which must be at least 1001 days (143 weeks) after the claim start date.
            2. There may not be more than 10 claim events.
            3. The days between snapshots must be at least 30 days and not more than 100 days.
        viii. For assets which are inflationary and don't have a fixed max supply there must be a distribution periodically to meet the distribution percentage. The number of times a periodic distribution occurs to maintain the inflation percentage can be indefinite. The following must be specified when the announcement page is created:
            1. The first snapshot date and time.
            2. The claim start date which must be within 7 days of the snapshot date.
            3. The claim end date which must be at least 1001 days (143 weeks) after the claim start date.
            4. The number of days between snapshots.
                a. The number of days between snapshots must be at least 30 and not more than 100.
    g. The Licensee must also include the PDF version of the license file in the repo of the software distribution. The version of the license file included in the software distribution must be the same as the one that was notarized by the Licensee.
    h. The Licensee must send an email to admin@shardus.com at least 30 days in advance of the first snapshot date. The email should include:

    i.   The URL of the announcement page on the Licensee website

    ii.   The URL of the software distribution repository

i.   The Licensee must take a snapshot of all addresses holding Shardus token (ULT) on the scheduled date and time as specified on the announcement page. There may be multiple snapshots over time.

    i.   The snapshot must include the addresses and the balance it had at the time of the snapshot.

    ii.   The snapshot must cover all networks where ULT is present. These currently include the Ethereum and Polygon network.

    iii.   The snapshot must be at the same time across all the networks where ULT is present.

    iv.   For an address that has ULT on multiple networks the sum of the ULT on all networks must be used.

    v.   The following addresses on all networks must not be included in the snapshot:

        1.   Issuing address: 0x19caf17b4EA9f8DD9B5E8f17Ab0C3C10F132691D

        2.   Polygon bridge address: 0x40ec5b33f54e0e8a33a975908c5ba1c14e5bbbdf

    vi.   Addresses belonging to exchanges, AMM and other contracts must also be excluded. For example:

        1.   0x061a7abe5313b886aa27fcc1f1c876fedf3ce1ea (SushiSwap - Polygon)

        2.   0x2A35F755c9938c63141FB8D26a9045ABE6917d71 (DODO - Polygon)

        3.   0x4B658c395804f90D76aA1995391E4730C7208dE7 (DODO - Polygon)

        4.   0xae70AE3D5197729bD1Be63415d40Ab251A89EBEb (Uniswap - Ethereum)

j.   The Licensee must deploy a claim smart contract and a claim server to process claims.

    i.   The smart contract must be on a well established network that has low transaction fees. For example the Shardeum network, but if this network is not ready then another network like Polygon can be used.

        1.   If the Licensee network has smart contract capabilities the contract to process claims can be deployed there.

    ii.   A loadSnapshot function on the smart contract is used to load the Merkel root of the list of addresses and the corresponding balances as they were when the snapshot was taken. The actual list is published on the claim server operated by the Licensee.

    iii.   A web page which allows users to claim assets on the Licensee network and check claim status is setup on the Licensee website where:

        1.   Users can connect their wallet to the web page.

        2.   Users can enter their claim address to see the amount of coins they can claim and if a claim has already been made or not.

            a.   This is checked using the claim server operated by the Licensee.

        3.   Users can enter the claim address and the recipient address on the Licensee network and click a Claim button to initiate the claim process.

        4.   When the user clicks on the Claim button

            a.   A transaction is submitted to a claim server and includes:

                i.   the claim address holding the ULT

                ii.   the recipient address on the network of the Licensee

                iii.   the ULT balance held by the claim address

                iv.   the amount being claimed on the Licensee network

                v.   the transaction is also signed and proves ownership of the address holding the ULT

            b.   This is first checked on the claim server which returns the Merkel proof if the claim is valid. The Merkel proof shows how the Merkel root can be generated starting with the claim address and balance.

            c.   A transaction is submitted to the claim smart contract which adds the Merkel proof to the above transaction.

            d.   The smart contract validates the Merkel proof and accepts the transaction if the proof is valid. It simply records the valid claim transactions and provides proof of what addresses made a claim.

        5.   The claim server operated by the Licensee monitors the smart contract for claim transactions and initiates a transaction on the network of the Licensee to fulfill the claim.

<ol type="a" start="1">
If the Licensee network has smart contract capabilities the fulfillment of the claim can be handled by the same smart contract that receives the claim request.

<ol type="i" start="4">
The actual distribution of the asset to those who have claimed must be completed within 24 hours of the claim.
Any unclaimed assets after the claim period ends do not need to be distributed by the Licensee to ULT holders, but must be burned after the claim end date. This is to ensure that there is no incentive for the Licensee to not distribute the assets.

<ol type="a" start="11">
<li value="11">The public announcement on the Licensee website and the activity on the smart contract provides proof of fulfilling the requirements to obtain a license.

<ol start="13">
Purchase License
<ol type="a">
This only applies to private networks or public networks without an asset.
The smart contract for the Shardus token on the Ethereum network has a "purchaseLicense" function. https://etherscan.io/address/0x09617f6fd6cf8a71278ec86e23bbab29c04353a7#code
The Licensee can query the contract to see what the current cost is in terms of Shardus tokens.
The cost will be some percentage of the current token circulating supply.
The Licensee can acquire Shardus tokens on the market.
The Licensee sends the required amount in Shardus tokens to the purchaseLicense function along with a string to identify the company and the network; such as "Company A Network X".
A license token will be sent to the address that sent the Shardus tokens and the Shardus tokens used in the purchase will be burned to reduce the total circulating supply.
The license token will serve as proof on the blockchain of fulfilling the requirements to obtain a license.
The license token is not transferable.